```
# Import required libraries
                                          It Make hardistions
   prom skleam import datasets to be and allowed
   from sklewn · model_selection import train_test_split
  from Sklewn - preprocessing impost Standard Scales
  from sklearn-sum import SVC
 from Sklearn. metrics import classification-report, confusion-matrix
   impost mathlotlibopyplot as plt babbard amos valgais #
#Load MNIST digits dataset ((4,01) = seiz gil) ampli all
  digits = datasets · load - digits () not bear again) xabon sol
  #Features and labels
                         hit. subflot (2,4, index+1)
  X = digits/data
  y = digits. tanget (8,8) stoders . 2 goal) works on . Ho
#split into training & testing sets
  X train, X test, y train, y test = train test split (x, y, test size
  =0.2, random-state=42, stratify=y)
 # Feature Scaling
 scales = Standard Scales ()
 X_train_scaled = scales. fit _transform (x-train)
 X_test_scaled = Scales - toansform (X_test)
# Train the SVM model
 sum_model = suc (kennel=(86), gamma = 0.001, c=10)
  SVm _model - fit (x-train _ scaled, y-train)
```

```
6 # Sofort required libraries
     # Make predictions
      y- fred = sum_model . foredict (x_test - scaled)
      #Evaluation of took of infort trains took of the social substances
      print ("Classification Report: \n", classification - seport (y-test,
        y-pred))
                                Loom stleam. sum impost SVC
   print (" Confusion Matrix: \n", confusion_matrix (y-test, y-pred))
      # Display Some predicted digits today altout me today
      hit. figure (fig size = (10,4)) texto still trum beal to
      for index, (image, prediction) in enumerate (zip (x_test[:8],
       9-pred [:8]):
                                          # Features and labels
            hlt. subplot (2,4, index+1)
                                              X = ofgits adda
           plt · im show (image · reshape (8,8), cmap = plt · cm · gray_s)
           plt . title (6" Pred: EP rediction 3")
                                    #split into training & testing
           plt - axis ('obb')
but tight lay out () = test - y anost - y lest x anost x
       hit. show ()
                                               11 Featons Scoling
                                       scolex = Standard Scoler()
            X train scaled = scales fit - tomsform (x train)
                     X = lat - scaled = Scales - transform (X tat)
                                     # Tools the SUM model
```

(01=2/1000 = emond (,19R) = Paries, ) ans = 1 apour - wis Stm \_ model - 12+ (X - Train - scaled , y-toon)

Memorat Mighbors (MIND: dosely the Tris datast

6th prg output

## Classification Report:

)

precision recall f1-score support

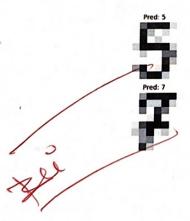
0	1.00	1.00	1.00	36
1	0.95	0.97	0.96	36
2	1.00	1.00	1.00	35
3	1.00	1.00	1.00	37
4	0.97	1.00	0.99	36
5	1.00	1.00	1.00	37
6	1.00	1.00	1.00	36
7	0.97	1.00	0.99	36
8	1.00	0.94	0.97	35
0	1.00	0.07	0.00	26

0.99 360 macro 0.99 0.99 0.99 360 weighted avg 0.99 0.99 0.99 360

## Confusion Matrix:

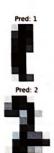
[[36 0 0 0 0 0 0 0 0 0] [035 0 0 1 0 0 0 0 0] [0 0 35 0 0 0 0 0 0 0] [00037000000] [0 0 0 0 36 0 0 0 0 0] [0 0 0 0 0 0 36 0 0 0] [00000003600] [0 2 0 0 0 0 0 0 33 0]

[00000001035]]









X\_train = Scalor-lit - transform (X-train) X\_test = scalex. Transform (x-test)

the create the KNN chasilies (you can change

as reeded)